



COMP2000

Object-Oriented Programming Practices

Session 2, In person-scheduled-weekday, North Ryde 2022

School of Computing

Contents

<u>General Information</u>	2
<u>Learning Outcomes</u>	2
<u>General Assessment Information</u>	3
<u>Assessment Tasks</u>	3
<u>Delivery and Resources</u>	4
<u>Unit Schedule</u>	6
<u>Policies and Procedures</u>	8

Disclaimer

Macquarie University has taken all reasonable measures to ensure the information in this publication is accurate and up-to-date. However, the information may change or become out-dated as a result of change in University policies, procedures or rules. The University reserves the right to make changes to any information in this publication without notice. Users of this publication are advised to check the website version of this publication [or the relevant faculty or department] before acting on any information in this publication.

General Information

Unit convenor and teaching staff

Convenor and Lectuer

Damian Jurd

damian.jurd@mq.edu.au

Lecturer

Ansgar Fehnker

ansgar.fehnker@mq.edu.au

Credit points

10

Prerequisites

COMP1010 or COMP125

Corequisites

Co-badged status

Unit description

Object-oriented programming is a key technology for modern computing. This unit bridges the gap between introductory programming and larger multi-person projects by considering the use of object-oriented techniques to produce intermediate sized software. Practical exercises emphasise the importance of programming practices such as appropriate documentation, systematic approaches to debugging and testing, and the use of software development tools. The unit is taught using Java.

Important Academic Dates

Information about important academic dates including deadlines for withdrawing from units are available at <https://www.mq.edu.au/study/calendar-of-dates>

Learning Outcomes

On successful completion of this unit, you will be able to:

ULO1: explain the key concepts of object oriented programming, and program proficiently in an OO programming language

ULO2: apply the concepts underlying software design and a working knowledge of a selection of well known design patterns

ULO3: demonstrate good programming practices such as testing, debugging,

documentation, version control, programming tools and interactive development environments

ULO4: apply key object oriented concepts and libraries to design and develop applications of significant complexity

ULO5: apply key concepts of concurrency theoretically and in working code

General Assessment Information

Late assessments are not accepted in this unit unless a Special Consideration has been submitted and approved.

Assessment Tasks

Name	Weighting	Hurdle	Due
Major Creative Work	60%	No	Weeks 5, mid-semester break, 11, and 12
Module Exams	40%	No	Weeks 6, 9, and 13

Major Creative Work

Assessment Type ¹: Programming Task

Indicative Time on Task ²: 40 hours

Due: **Weeks 5, mid-semester break, 11, and 12**

Weighting: **60%**

A semester-long programming task where students put all their skills to work creating a game or demo.

On successful completion you will be able to:

- explain the key concepts of object oriented programming, and program proficiently in an OO programming language
- apply the concepts underlying software design and a working knowledge of a selection of well known design patterns
- demonstrate good programming practices such as testing, debugging, documentation, version control, programming tools and interactive development environments
- apply key object oriented concepts and libraries to design and develop applications of significant complexity
- apply key concepts of concurrency theoretically and in working code

Module Exams

Assessment Type ¹: Examination

Indicative Time on Task ²: 16 hours

Due: **Weeks 6, 9, and 13**

Weighting: **40%**

A number of exams spread through the semester. Students will have the opportunity to repeat any exam to improve their mark.

On successful completion you will be able to:

- explain the key concepts of object oriented programming, and program proficiently in an OO programming language
- apply the concepts underlying software design and a working knowledge of a selection of well known design patterns
- demonstrate good programming practices such as testing, debugging, documentation, version control, programming tools and interactive development environments
- apply key object oriented concepts and libraries to design and develop applications of significant complexity
- apply key concepts of concurrency theoretically and in working code

¹ If you need help with your assignment, please contact:

- the academic teaching staff in your unit for guidance in understanding or completing this type of assessment
- the [Writing Centre](#) for academic skills support.

² Indicative time-on-task is an estimate of the time required for completion of the assessment task and is subject to individual variation

Delivery and Resources

CLASSES

Each week of COMP2000 has up to two hours of online lectures and a two-hour practical class. For details of days, times and rooms, consult the University timetables webpage (<http://www.time-tables.mq.edu.au>). Practical classes commence in Week 1 and are held in the 4RPD Computer Laboratories computer laboratories for on-campus classes and in zoom rooms for online classes (links published in iLearn).

Students choosing online practicals are expected to have camera, microphone, and screen sharing capabilities for all classes. If you don't have access to those, please choose an on-campus class.

In all cases students are expected to do significant preparatory work, readings and exercises *before* attending classes.

REQUIRED AND RECOMMENDED TEXTS AND/OR MATERIALS

The required text for the unit is (available online via the Macquarie University Library, see below):

- Head First Design Patterns by Eric T Freeman, Elisabeth Robson, Bert Bates and Kathy Sierra, O'Reilly Media, October 2004 (ISBN:978-0-596-00712-6)

There will be no lecture notes provided, all examinable material is given in course readings and the textbook. Students are required to study this material and answer preparatory questions *before* class.

The Macquarie library contains many books on object-oriented programming in general, and on programming specifically in Java, that you may want to use to supplement the text and lecture notes.

One particularly useful service that the library provides you with is access to many Java related titles online via the Safari Books Online (<http://proquest.safaribooksonline.com/>) service. Using this service, which you can only access from a machine connected to the University network, you might like to have a look at the following Java titles:

1. Head First Design Patterns by Eric T Freeman, Elisabeth Robson, Bert Bates and Kathy Sierra, O'Reilly Media, October 2004 (ISBN:978-0-596-00712-6)
2. Learning Java, 3rd Edition by Jonathan Knudsen; Patrick Niemeyer, ISBN: 978-0-596-00873-4
3. Java in a Nutshell, 5th Edition by David Flanagan, ISBN: 978-0-596-00773-7
4. Java Examples in a Nutshell, 3rd Edition by David Flanagan, ISBN: 978-0-596-00620-4
The web itself is an ideal source of Java information, and from time to time we will be posting useful links on the COMP2000 iLearn site. Two particularly useful resources are:
5. Thinking in Java by Bruce Eckel, a free version of the 3rd edition of this pretty comprehensive book is available for download from <http://www.mindview.net/Books/TIJ/> and its 4th edition, which is updated for use with Java 5 and 6, is available for \$25 from <http://mindview.net/Books/TIJ4>.
6. The official Java Tutorial <http://download.oracle.com/javase/tutorial/> which is a comprehensive resource providing trails covering topics ranging from the basics of Java programming to more advanced subjects like GUI development, Generics, Class Reflection, Sound, Graphics, Network Programming and Concurrency

UNIT WEBPAGE AND TECHNOLOGY USED AND REQUIRED

Online Resources

COMP2000 will make extensive use of the iLearn system for delivery of class materials, discussion boards, real time chat, submission of work and access to marks and feedback. Students should check the iLearn site (<http://ilearn.mq.edu.au>) regularly for unit updates.

Questions and general queries regarding the content of this unit, its tutorials or practicals should be posted to the appropriate discussion board on the COMP2000 iLearn site. In particular, any questions which are of interest to all students in this unit should be posted to one of these discussion boards, so that everyone can benefit from the answers.

Echo360

Audio and screen video recordings of the lectures will be made available online via Echo360. A link to these recordings will be provided on the iLearn site for this unit

Technology Used and Required

The practical work in this unit involves programming in Java (<https://www.java.com>) using the Microsoft Visual Studio Code IDE (<https://code.visualstudio.com>). We will also be using a distributed version control system called Git to access shared code repositories hosted on the BitBucket website (<https://bitbucket.org>).

This software is already installed for you in the computing labs located in 4 Research Park Drive and is also available to download, install and use for free on your own machine(s). It should work equally well on Mac OSX, Linux or Microsoft Windows platforms.

Tools and libraries to support debugging, automated testing, GUI development and so forth will be introduced and used as the semester progresses. When that occurs you will be provided with full instructions in lectures on how to install and use each one.

Unit Schedule

Tentative teaching schedule, subject to change:			Assessment		Workshops	
Week	Lecture	Lecturer	Due	Weight	In-class activity	Exam topics
1	Version Control	DJ			Version Control	
2	Java Classes and Objects	DJ			Java Classes and Objects	
3	Inheritance and Overloading	AF			Inheritance and Overloading	

4	Generics	DJ			Generics	
5	Exceptions	DJ			Exceptions and Lambdas	Version Control / Exercise Class / Late Binding / Reference vs Value / Overloading vs Overriding
	Lambda Expressions		Project Milestone 1	10%		Reference vs Value / Overloading vs Overriding / Generics / Advanced Generics
6	Introduction to Patterns, Strategy Pattern, Observer Pattern	AF			Strategy Pattern, Observer Pattern	
			Exam : Java Concepts	15%	Exam: Java Concepts	
7	Decorator Pattern, Iterator/Composite Patterns	AF			Decorator Pattern, Iterator/ Composite Patterns	
	Break week 1					
	Break week 2		Project Milestone 2	10%		
8	Adapter and Facade Patterns	AF			Adapter and Facade Patterns	
9	Singleton Pattern, Command Pattern	AF			Singleton Pattern, Command Pattern	
			Exam: Design Patterns	15%	Exam: Design Patterns	
10	Concurrency	DJ			Concurrency	Lambdas, Exceptions / Classify Patterns, Name that Pattern / Complete Intent of Design Pattern, Shadow Class Diagram

						Design Principles / Pattern Hypothetical / (Applying Patterns) / Concurrency terms
11	Concurrency	DJ			Concurrency	
			Project Milestone 3	10%		
12	Concurrency	DJ	Viva Exam	30%	Viva Exam	
13	Review	AF				
			Exam: Concurrency	10%	Exam: Concurrency	Basics of Java Threading / Making and using threads / Thread safety / Advanced thread topics
14	Formal Exam Period		Module Exams		Second attempt at module exams	
15						
16						

Policies and Procedures

Macquarie University policies and procedures are accessible from [Policy Central \(https://policies.mq.edu.au\)](https://policies.mq.edu.au). Students should be aware of the following policies in particular with regard to Learning and Teaching:

- [Academic Appeals Policy](#)
- [Academic Integrity Policy](#)
- [Academic Progression Policy](#)
- [Assessment Policy](#)
- [Fitness to Practice Procedure](#)
- [Assessment Procedure](#)
- [Complaints Resolution Procedure for Students and Members of the Public](#)
- [Special Consideration Policy](#)

Students seeking more policy resources can visit [Student Policies \(https://students.mq.edu.au/support/study/policies\)](https://students.mq.edu.au/support/study/policies). It is your one-stop-shop for the key policies you need to know about throughout your undergraduate student journey.

To find other policies relating to Teaching and Learning, visit [Policy Central \(https://policies.mq.edu.au\)](https://policies.mq.edu.au) and use the [search tool](#).

Student Code of Conduct

Macquarie University students have a responsibility to be familiar with the Student Code of Conduct: <https://students.mq.edu.au/admin/other-resources/student-conduct>

Results

Results published on platform other than [eStudent](#), (eg. iLearn, Coursera etc.) or released directly by your Unit Convenor, are not confirmed as they are subject to final approval by the University. Once approved, final results will be sent to your student email address and will be made available in [eStudent](#). For more information visit ask.mq.edu.au or if you are a Global MBA student contact globalmba.support@mq.edu.au

Academic Integrity

At Macquarie, we believe [academic integrity](#) – honesty, respect, trust, responsibility, fairness and courage – is at the core of learning, teaching and research. We recognise that meeting the expectations required to complete your assessments can be challenging. So, we offer you a range of resources and services to help you reach your potential, including free [online writing and maths support](#), [academic skills development](#) and [wellbeing consultations](#).

Student Support

Macquarie University provides a range of support services for students. For details, visit <http://students.mq.edu.au/support/>

The Writing Centre

[The Writing Centre](#) provides resources to develop your English language proficiency, academic writing, and communication skills.

- [Workshops](#)
- [Chat with a WriteWISE peer writing leader](#)
- [Access StudyWISE](#)
- [Upload an assignment to Studiosity](#)
- [Complete the Academic Integrity Module](#)

The Library provides online and face to face support to help you find and use relevant information resources.

- [Subject and Research Guides](#)
- [Ask a Librarian](#)

Student Services and Support

Macquarie University offers a range of [Student Support Services](#) including:

- [IT Support](#)
- [Accessibility and disability support](#) with study

- Mental health [support](#)
- [Safety support](#) to respond to bullying, harassment, sexual harassment and sexual assault
- [Social support including information about finances, tenancy and legal issues](#)

Student Enquiries

Got a question? Ask us via [AskMQ](#), or contact [Service Connect](#).

IT Help

For help with University computer systems and technology, visit http://www.mq.edu.au/about_us/offices_and_units/information_technology/help/.

When using the University's IT, you must adhere to the [Acceptable Use of IT Resources Policy](#). The policy applies to all who connect to the MQ network including students.